

Seminarvortrag

Java Studio Creator 2

von

Björn Nebe



Übersicht

- JSC – Was ist das?
- Download und Installation
- Erstellen von WebApplications
- Erstellen von Portlets
- Ausblick: weitere Funktionen
- Livedemo
- Fazit



JSC – Was ist das ?

- Java-Entwicklungsumgebung für Webanwendungen.
- Ermöglicht das Erstellen komplexer Webseiten-Strukturen, deren Navigation, Datenbankbindung, Einbinden von Ajax oder Webservices und vieles mehr durch einfaches Drag & Drop.
- Die im Paket enthaltenen Server simulieren im Hintergrund alle benötigten Umgebungen.



Download und Installation

- Das Studio lässt sich bei Sun kostenlos downloaden, allerdings ist zuvor eine Anmeldung erforderlich.
- Verfügbar für Windows, Linux, Mac OS und Solaris
- Aktuelle Version: Java Studio Creator 2 Update1
- Installation per Wizard

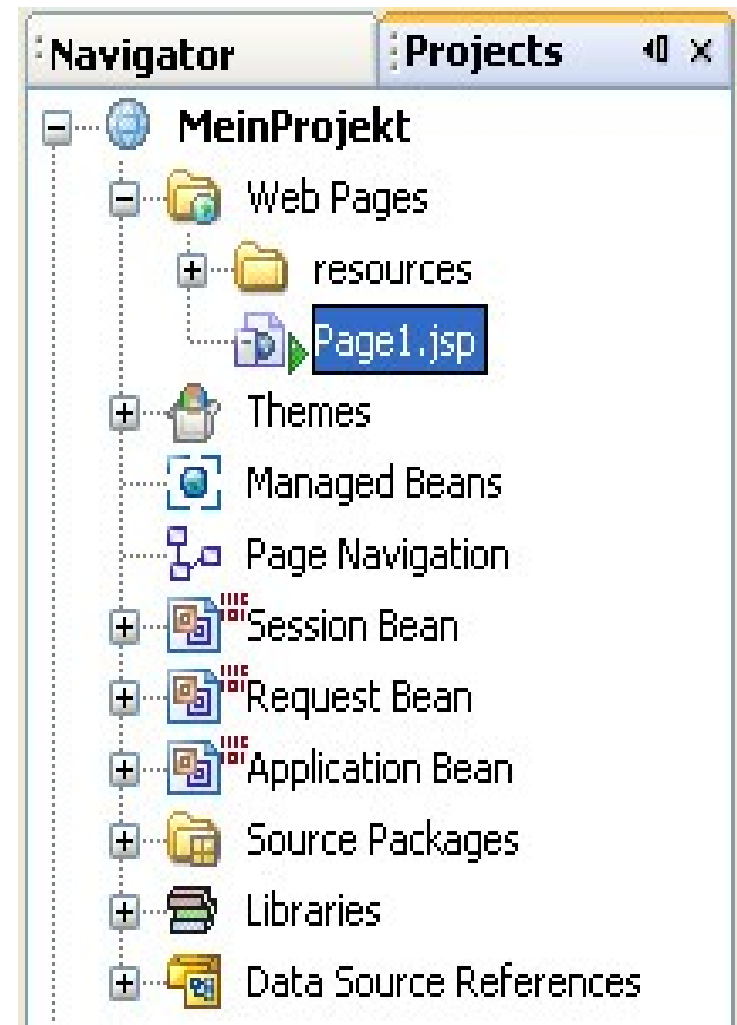




Erstellen von WebApplications

neues Projekt

- Zunächst erstellt man mit „new Projekt“ eine JSF Web Application. Dadurch wird rechtstehende Stuktur erzeugt.

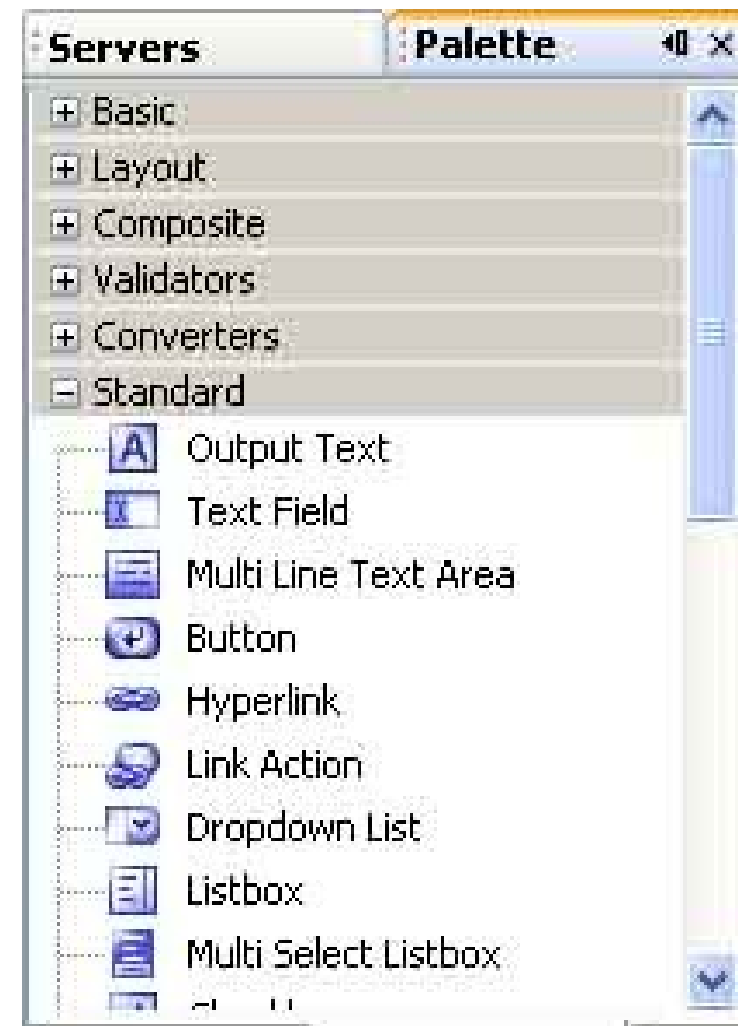




Erstellen von WebApplications

Elemente

- Nun können aus der Palette beliebige Elemente ausgewählt und per Drag & Drop angeordnet werden.






Erstellen von WebApplications


Arbeitsoberfläche - Design

Page1 * x

Design JSP Java Any Size

Eingabe:


dd.mm.yyyy

abc 

zuSeite2



Erstellen von WebApplications

Arbeitsoberfläche - JSP

```
Page1 * x
Design JSP Java
1 <?xml version="1.0" encoding="UTF-8"?>
2 <jsp:root version="1.2" xmlns:f="http://java.sun.com/jspf/core" xmlns:h="http://
3   <jsp:directive.page contentType="text/html; charset=UTF-8" pageEncoding="UTF
4   <f:view>
5     <ui:page binding="#{Pagel.pagel}" id="pagel">
6       <ui:html binding="#{Pagel.html1}" id="html1">
7         <ui:head binding="#{Pagel.head1}" id="head1">
8           <ui:link binding="#{Pagel.link1}" id="link1" url="/resource
9         </ui:head>
10        <ui:body binding="#{Pagel.body1}" id="body1" style="-rave-layou
11        <ui:form binding="#{Pagel.form1}" id="form1">
12          <ui:label binding="#{Pagel.eingabelabel}" for="eingabef
13            style="height: 22px; left: 48px; top: 48px; positio
14          <ui:textField binding="#{Pagel.eingabefeld}" id="eingab
15          <ui:calendar binding="#{Pagel.calendar1}" id="calendar1
16          <ui:dropDown binding="#{Pagel.dropDown1}" id="dropDown1
17          <ui:button binding="#{Pagel.zuseite2}" id="zuseite2" st
18        </ui:form>
19      </ui:body>
20    </ui:html>
```




Erstellen von WebApplications

Arbeitsoberfläche - Java

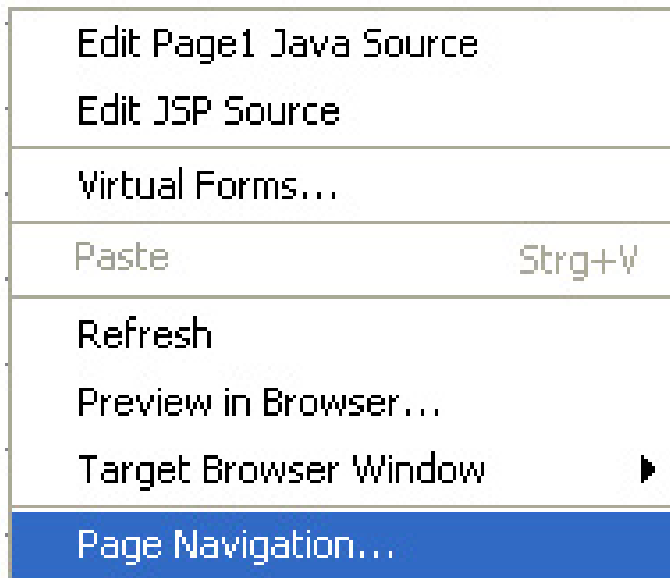
```
174  */
175  protected ApplicationBean1 getApplicationBean1() {
176      return (ApplicationBean1) getBean("ApplicationBean1");
177  }
178
179
180  /**
181   * <p>Return a reference to the scoped data bean.</p>
182   */
183  protected RequestBean1 getRequestBean1() {
184      return (RequestBean1) getBean("RequestBean1");
185  }
186
187
188  /**
189   * <p>Return a reference to the scoped data bean.</p>
190   */
191  protected SessionBean1 getSessionBean1() {
192      return (SessionBean1) getBean("SessionBean1");
193  }
```



Erstellen von WebApplications

Page Navigation

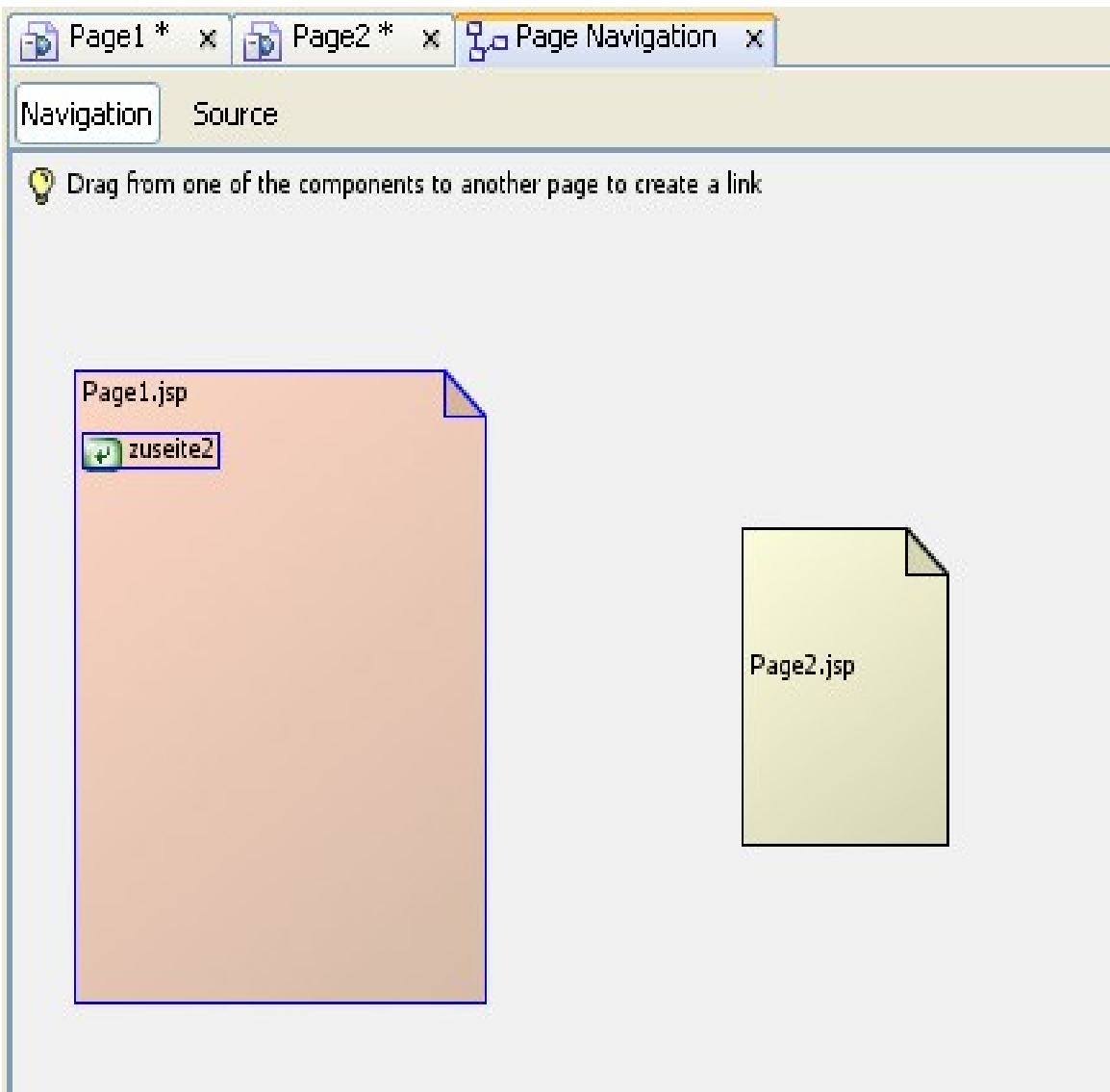
- Per Rechtsklick in die Design-Ansicht und klicken auf „Page Navigation“ gelangt man in die Navigationsansicht



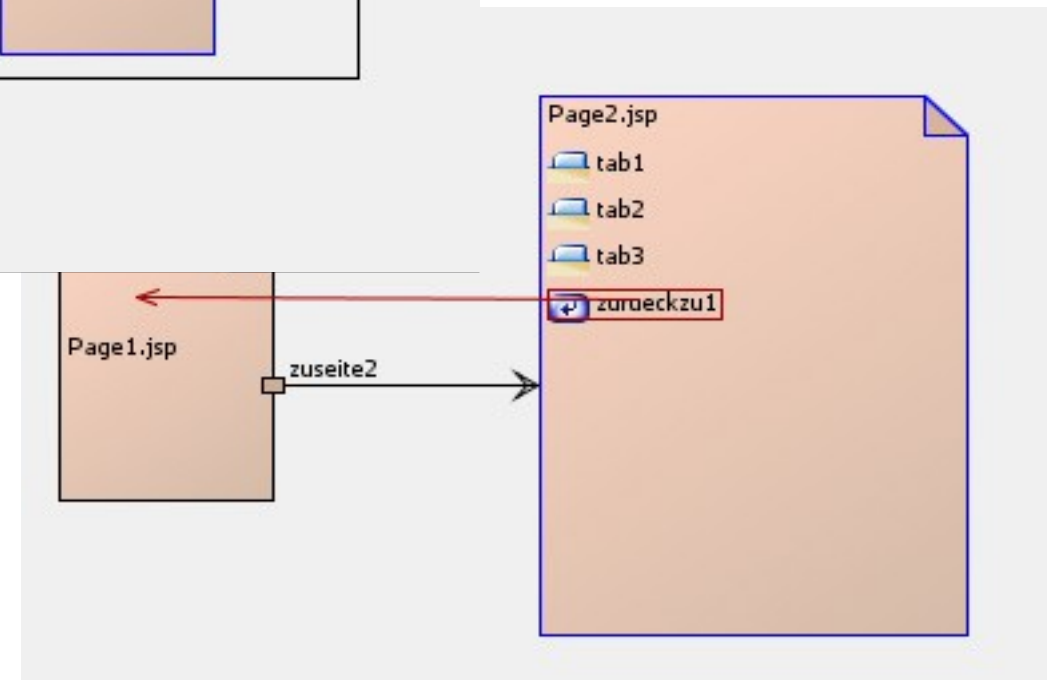
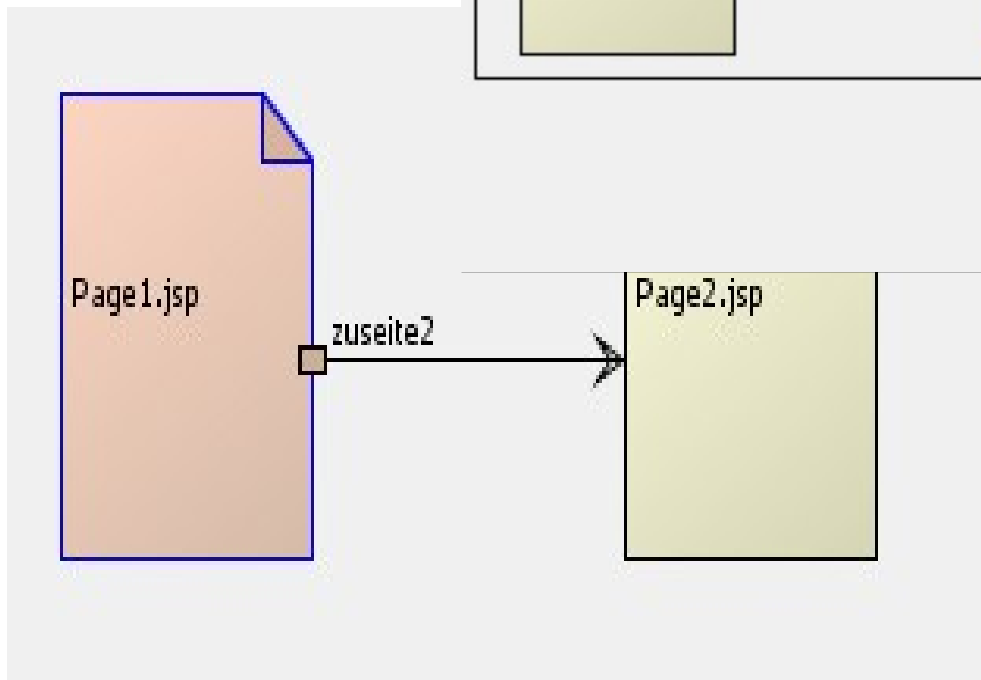
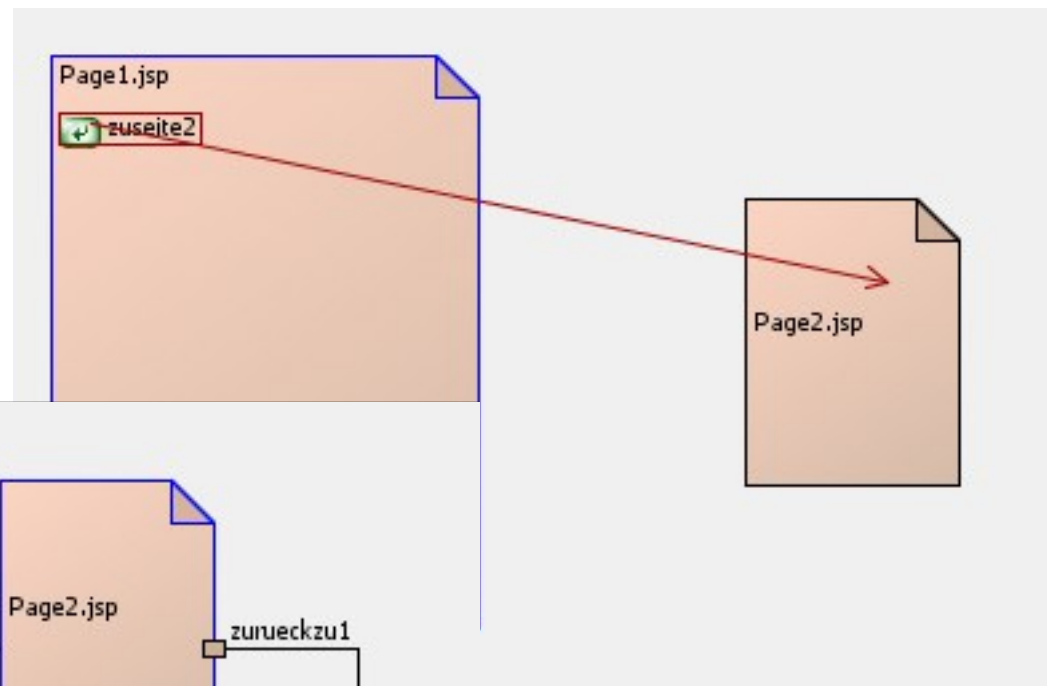
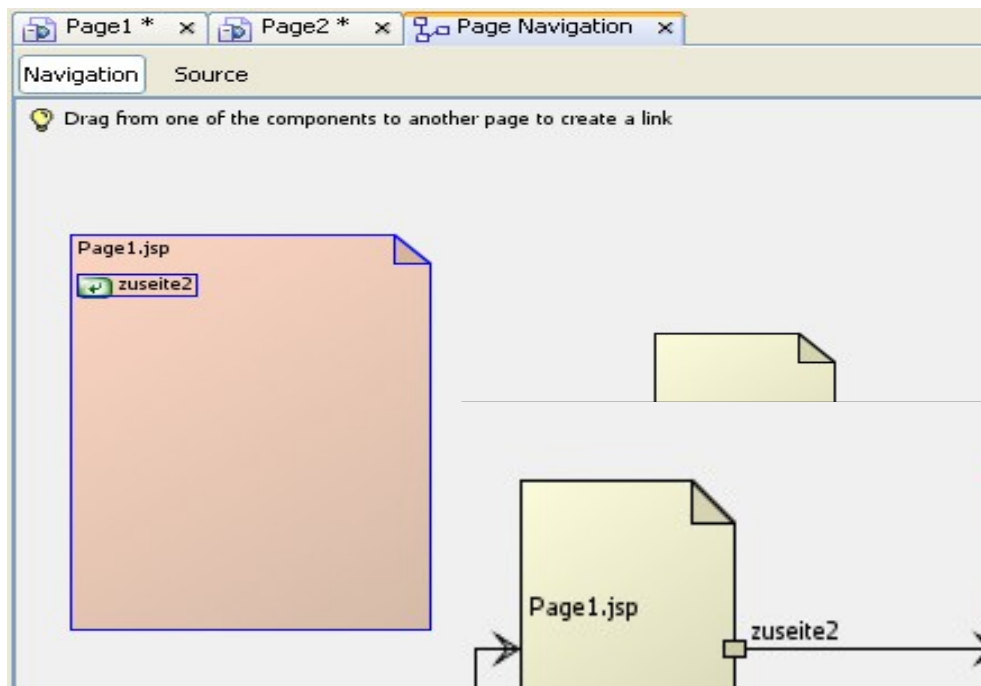


Erstellen von WebApplications

Page Navigation



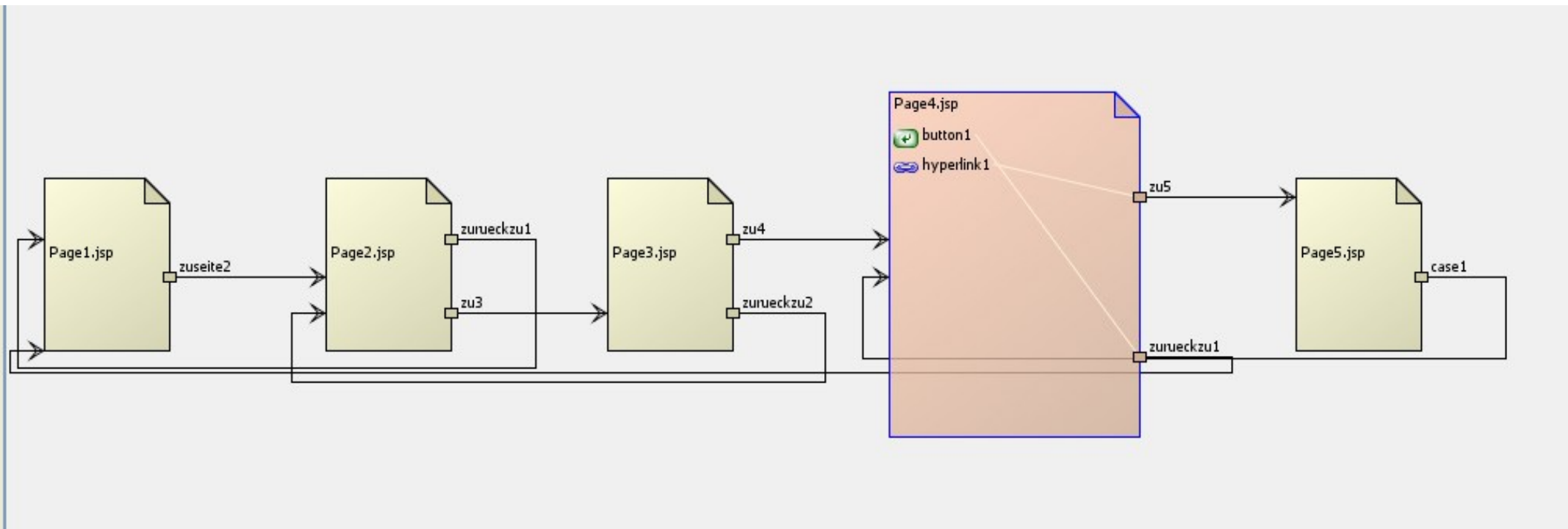
- Alle Seiten und deren Links und Buttons in der Übersicht
- Hier zunächst mit 2 Seiten, die jeweils einen Button beinhalten





Erstellen von WebApplications

Page Navigation



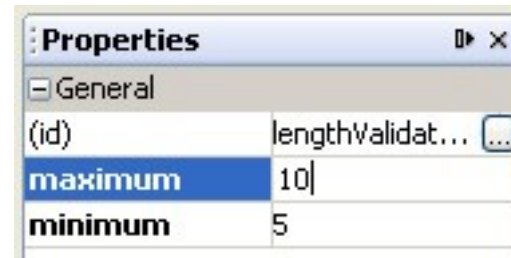
- Mit mehr Seiten und Links sieht unser Projekt schließlich so aus



Erstellen von WebApplications

Funktionalität hinzufügen

- Bsp.: Validierer
 - Validierer auf Textfeld ziehen
 - Wertebereich eingeben

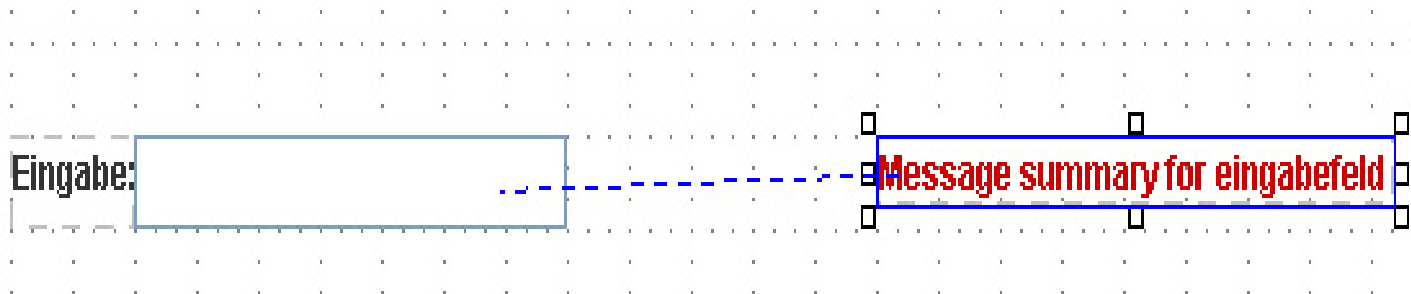




Erstellen von WebApplications

Funktionalität hinzufügen

- Bsp.: Validierer
 - Validierer auf Textfeld ziehen
 - Wertebereich eingeben
- Message zum Informieren des Users einfügen





Erstellen von WebApplications

Funktionalität hinzufügen

- Bsp.: Validierer
 - Validierer auf Textfeld ziehen
 - Wertebereich eingeben
 - Message zum Informieren des Users einfügen
 - Ergebnis im Browser bewundern



Eingabe:

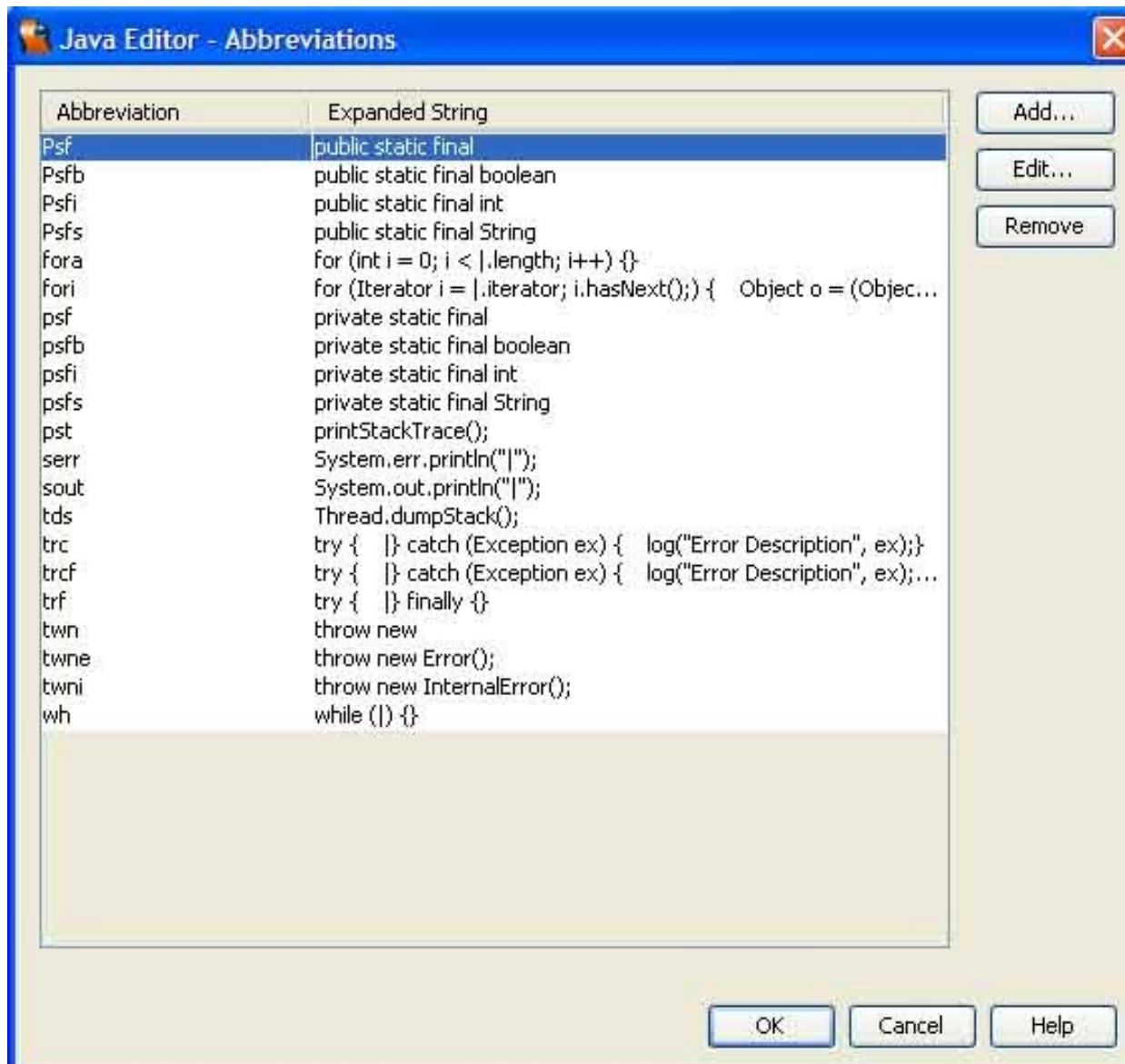
meinzulangespaword

Überprüfungsfehler: Wert ist größer als der zulässige Maximalwert '10'.



Erstellen von WebApplications

Programmierhilfen

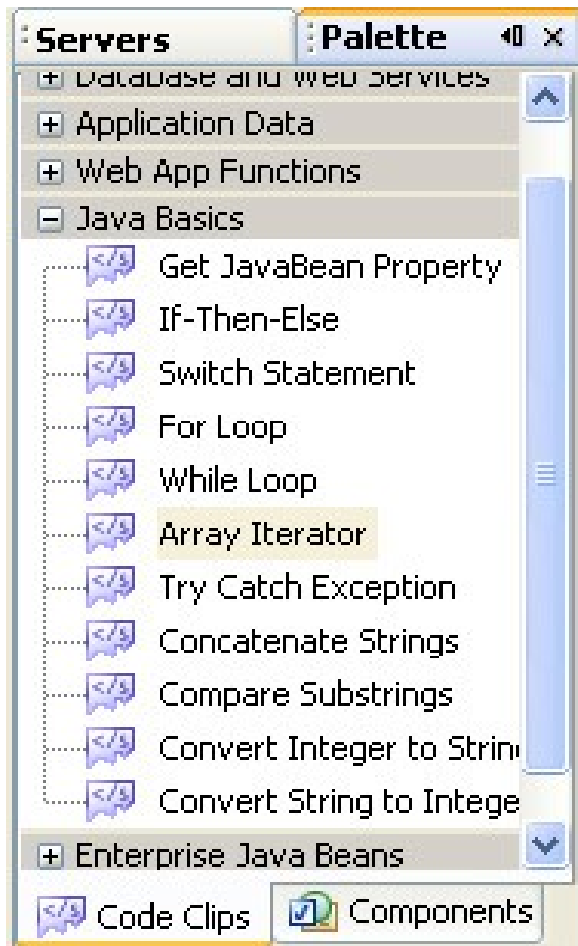


- Abbreviations
 - Eingeben einer Abkürzung
 - Drücken von *Enter*



Erstellen von WebApplications

Programmierhilfen



```
// Create a list of items
java.util.ArrayList popItems = new ArrayList();
popItems.add("Hammerhead");
popItems.add("Thresher");
popItems.add("Great White");

// Initialize the iterator and other variables
java.util.Iterator items = popItems.iterator();
String item;
getTextArea1().setText("My favorite sharks: ");

// Loop through the items, adding each to the text area
while (items.hasNext()) {
    item = (String)items.next();
    getTextArea1().setText(getTextArea1().getText() + " " + item);
} // end while
```



Erstellen von WebApplications

Programmierhilfen

- Codevervollständigung und Imports verwalten
 - Natürlich sind auch hier die üblichen Funktionen implementiert:
 - Strg + Leertaste um alle Möglichkeiten zur Vervollständigung oder alle Methoden, über die dieses Objekt verfügt, anzuzeigen.
 - Rechtsklick->Fix Imports um fehlende Imports vorzunehmen oder überflüssige zu entfernen
 - Formatieren etc....



20/34



Erstellen von WebApplications

Programmierhilfen

- Der Debugger
 - Standardfunktionalität: Breakpoints
 - Watches: eine Variable überwachen lassen. Alle Watches sind in einem eigenen Bereich aufgelistet.
 - HTTP-Monitor: Requests erneut ausführen lassen. Das erspart bei mehrmaligem Testen das erneute Ausfüllen der Formulare.
 - Einträge in Logs und MessageGroups



Erstellen von WebApplications

Speichern von Daten

- Beans
 - Request-Beans
 - Daten eines einzelnen Requests speichern.
 - Beispiel: Auswertung von Benutzereingaben in einem Formular und eine darauf generierte Antwort.
 - Request-Bean-Objekt wird unmittelbar nach dem Erstellen der Antwort gelöscht und die Informationen gehen, sofern nicht verwertet, verloren.



Erstellen von WebApplications

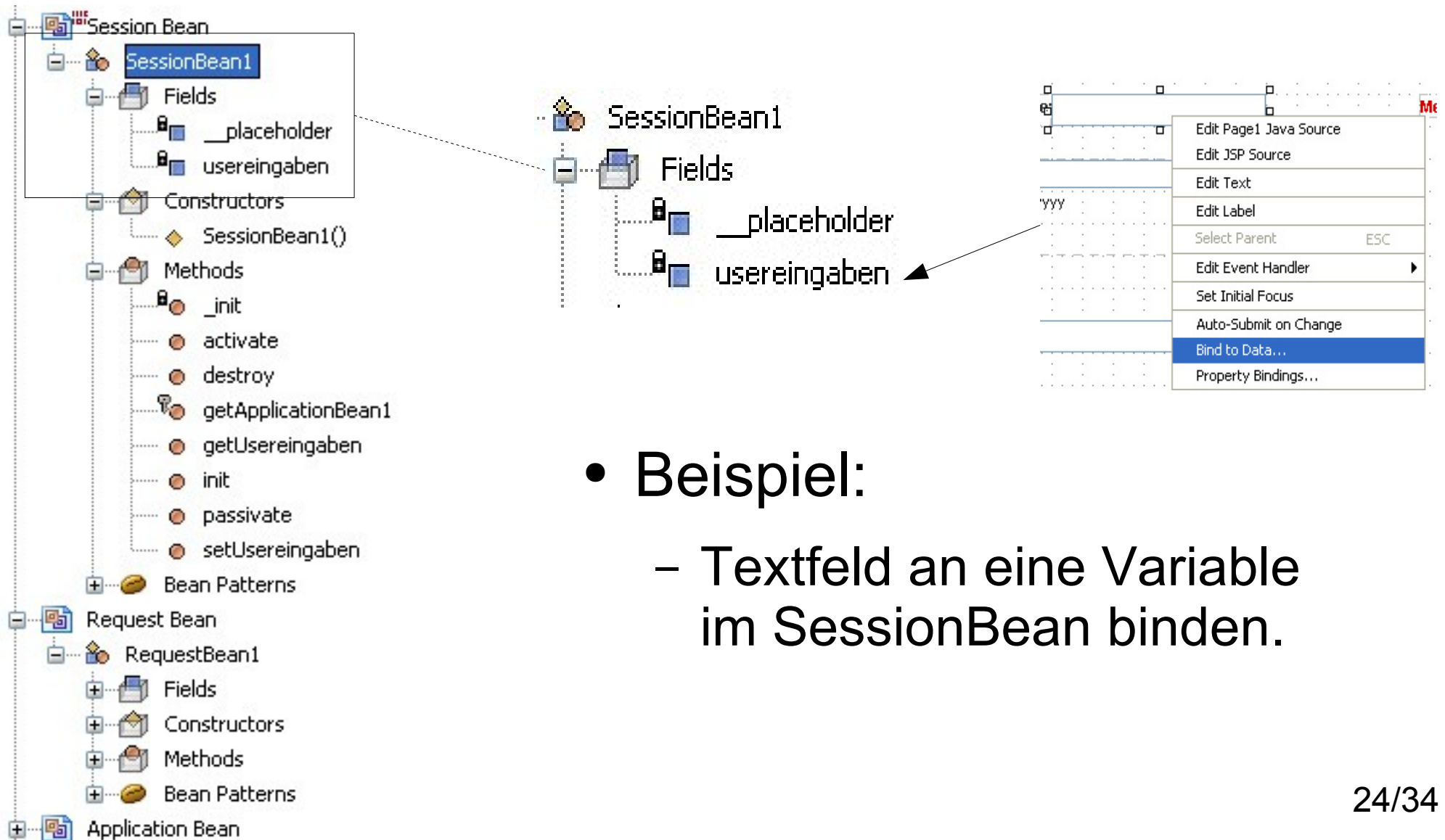
Speichern von Daten

- Beans
 - Session-Beans
 - Speichern die Information eines Benutzers über die komplette Dauer seiner Session, also über die komplette Zeit, die der Benutzer sich auf der Seite aufhält.
 - Application-Beans
 - Ein Application-Bean speichert Information, die für jede Session jedes Benutzers von Bedeutung sind.



Erstellen von WebApplications

Speichern von Daten



- Beispiel:
 - Textfeld an eine Variable im SessionBean binden.



Erstellen von WebApplications

Speichern von Daten

- Datenbanken
 - Natürlich können Komponenten auch direkt an eine Datenbank gebunden werden, dazu muss diese allerdings erst konfiguriert werden.
 - ->Mehr dazu siehe Wiki-Artikel
 - Binden dann ebenfalls per „Bind to Data“ wie beim Binden an Beans



Erstellen von Portlets

- Die zweite Art von Projekten in JSC sind Portlets.
- Die Spezifikation JSR-168 Standard legt drei Portlet-Modi fest:
 - View-Modus
 - Edit-Modus
 - Help-Modus



Erstellen von Portlets

- Wikipedia:
 - Ein Portlet ist ein Programm, welches in der Programmiersprache Java geschrieben ist, und die Fähigkeiten eines Portalservers in derselben Weise erweitert, wie ein Servlet diejenigen eines Webserver.
 - Portlets bilden auf der Clientseite eine einfach zu benutzende Oberfläche innerhalb des Browsers (Fenster mit Schaltflächen zum Maximieren, Minimieren, Editieren, Hilfe). Intern, also auf Serverseite, kann nun eine beliebige Anwendung liegen, die ihre Darstellung auf das Portlet weiterleitet.



Erstellen von Portlets

Sun Java[tm] Studio Creator
Portlet Preview

Pluto Portal Driver (pluto-driver/1.0.1) deployed in Sun-Java-System/Application-Server

Portlet1

Portlet1

edit help view | max min nor

Willkommen auf Seite 1

Auf gehts zu Seite 2

The time is...



Erstellen von Portlets

- Hier zu beachten:
 - Startseiten der einzelnen Bereiche festlegbar
 - Beim Wechsel in einen anderen Bereich wird immer wieder dessen Startseite aufgerufen
 - Anderer Lebenszyklus der Objekte -> Request-Beans nicht nutzbar
 - Extra Portlet-Server vorhanden: Apache Pluto Portal Server



Ausblick: weitere Funktionen

- Nach dem Nachinstallieren der entsprechenden Komponenten ist Ajax nutzbar
- Unterstützung für Internationalisierung von Projekten (Ländercodierungen)
- Unterstützung von Seitenteilen („Page Fragments“), die dann in mehreren Seiten verwendet werden können



Ausblick: weitere Funktionen

- Virtuelle Formulare (Virtual Forms)
 - künstliche Bereiche innerhalb einer Seite
 - Jedes virtuelle Formular hat mindestens ein Element, das dem Ausführen des Formulars dient, und beliebig viele, frei wählbare weitere Komponenten
 - In JSC gibt es ein Menü, in dem die Zugehörigkeit zu virtuellen Formularen festgelegt werden kann



Ausblick: weitere Funktionen

- Projekt-Templates
 - Projekte können als Template gespeichert werden, so dass bei weiteren Projekten die Dateistruktur, die Style Sheets oder die Projekteinstellungen übernommen werden können
 - Sun empfiehlt, diesen Mechanismus als Sicherungskopie zu nutzen. Hierzu könnte man ab und zu eine laufende Version als Template speichern und dann damit als ein neues Projekt weiter arbeiten



Ausblick: weitere Funktionen

- Style Sheets
 - Es existieren diverse Wizards, um neue Style-Sheets zu erstellen, vorhandene zu ändern oder Komponenten einzelne Regeln hinzuzufügen
- JSC ist verknüpft mit Dateimanagement- und Versionierungstools wie CVS
- etc.



Fazit

- sehr mächtiges und allumfassendes Tool zum Entwickeln von Web-Applikationen in Java
- benutzerfreundlich gestaltet
- bleibt selbst bei komplexere Anwendungen immer recht übersichtlich
- aber:
 - langsam
 - ressourcenlastig



Quellen

- Online-Dokumentation von Sun
 - <http://developers.sun.com/prodtech/javatools/jscreator/>

Danke für die Aufmerksamkeit!

